AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1-41. (Cancelled)

42.   (Currently Amended) A computer-implemented method of managing a process, said computer implemented method comprising:

identifying activities that comprise the process, wherein the process is a series of activities, wherein an input of at least one subsequent activity is dependent on an output of at least one previous activity;

identifying drivers associated with at least one metric, reflecting an efficiency of said process, for each of the activities;

identifying, in a computer, bridge variables from said identified drivers, wherein each bridge variable is a driver that is relevant to more than one of said activities;

establishing, in a computer, a relationship between various drivers by representing at least one non-bridge variable driver in terms of one or more of said bridge variables;

using said relationship, representing activities at least as a function of one or more of said bridge variables, thereby reflecting interdependence between said activities to represent the entire series of activities of said process, in a computer;

generating, in a computer, a model of said process at least as a function of said bridge variables by combining representations of activities comprising said process; and

outputting, from said model in a computer, a predictive metric reflecting an efficiency of the total process.

43.　(Currently Amended)　The computer-implemented method of claim 42, further comprising:

selecting a plurality of constraints,

and wherein generating said model of said process includes generating said model as a function of said bridge variables and said plurality of constrains.

44.　(Previously Presented)　The computer-implemented method of claim 43, further comprising:

optimizing said model in view of said plurality of constraints using one of the following:

a linear programming algorithm,

a mixed-integer linear programming algorithm, and

a mixed-integer nonlinear programming algorithm; and

reconstructing a representation of said activities and said drivers using said optimized model.

45.　(Previously Presented)　The new computer-implemented method of claim 44, wherein said reconstructing includes calculating a value of at least one

non-bridge variable driver using values of corresponding bridge variables, and calculating a value of activities using values calculated for bridge variable driver and non-bridge variable driver of respective activities.

46.   (Previously Presented) The computer-implemented method of claim 44, further comprising:

revising said model using the results from said optimization step.

47.   (Previously Presented) The computer-implemented method of claim 43, wherein selecting said plurality of constraints includes setting economic and non-economic constraints.

48.   (Previously Presented) The computer-implemented method of claim 42, wherein identifying measurable drivers include identifying economic and non-economic drivers.

49.   (Previously Presented) The computer-implemented method of claim 42, wherein identifying said drivers includes identifying at least one of fixed and variable components of each said driver, and wherein said method further comprising:

determining said metric of each said driver based on said at least one of fixed and variable components thereof.

50.   (Previously Presented) A system comprising:

a computer;

input and output devices in communication with said computer; and

a memory encoded with a computer program, which, when executed by said computer, causes said computer to perform the following:

allow a user to identify activities that comprise a process, wherein the process is a series of activities, wherein an input of at least one subsequent activity is dependent on an output of at least one previous activity,

further allow said user to identify measurable drivers associated with a at least one metric, reflecting an efficiency of said process, for each of the activities; identify bridge variables from said identified drivers, wherein each bridge variable is a driver that is relevant to more than one of said activities; establish a relationship between various drivers by representing at least one non-bridge variable driver in terms of one ore more said bridge variables; using said relationship, represent activities at least as a function of one ore more of said bridge variables, thereby reflecting interdependence between said activities to represent the entire series of activities of said process; generate a model of said process at least as a function of said bridge variables by combining representations of activities comprising said process; and output, from said model, a predictive metric reflecting an efficiency of the total process.

51.   (Previously Presented) The system of claim 50, wherein said computer program, upon execution by said computer, causes said computer to further perform the following:

further allow said user to select a plurality of constraints;

incorporate said plurality of constraints in said model of said process;

optimize said model in view of said plurality of constraints using one of the following:

a linear programming algorithm,

a mixed-integer linear programming algorithm, and

a mixed-integer nonlinear programming algorithm; and

reconstruct a representation of said activities and said drivers using said optimized model.

52.   (Previously Presented) The system of claim 51, wherein said computer program, upon execution by said computer, causes said computer to perform said reconstruction by calculating a value of at least one non-bridge variable driver using values of corresponding bridge variables and by calculating a value of activities using values calculated for bridge variable drivers and non-bridge variable driver of respective activities.

53.   (Previously Presented) A computer-readable data storage medium containing program instructions, which, when executed by a processor, cause said processor to perform the following:

allow a user to identify activities that comprise a process, wherein the process is a series of activities, wherein an input of at least one subsequent activity is dependent on an output of at least one previous activity;

further allow said user to identify drivers associated with a metric

reflecting an efficiency of said process for each of the activities;

identify bridge variables from said identified drivers, wherein each

bridge variable is a driver that is relevant to more than one of said activities;

establish a relationship between various drivers by representing at

least one non-bridge variable driver in terms of one or more of said bridge variables;

using said relationship, represent activities at least as a function of one

or more of said bridge variables, thereby reflecting interdependence between said

activities to represent the entire series of activities of said process;

generate a model of said process at least as a function of said bridge

variables by combining representations of activities comprising said process; and

output, from said model, a predictive metric reflecting on efficiency of

the total process.

54.   (Previously Presented) The storage medium of claim 53, wherein said

program instructions, upon execution, cause said processor to further perform the

following:

further allow said user to select a plurality of constraints;

include said plurality of constraints in said model of said process; and

optimize said model in view of said plurality of constraints using one of

the following:

a linear programming algorithm,

a mixed-integer linear programming algorithm, and

a mixed-integer nonlinear programming algorithm; and

revise said model using the results from optimizing said model.

55.   (Previously Presented) The storage medium of claim 53, wherein said program instructions, upon execution, cause said processor to reconstruct a representation of said activities and said drivers by calculating a value of at least one non-bridge variable driver using values of corresponding bridge variables and by calculating a value of activities using values calculated for bridge variable driver and non-bridge variable driver of respective activities.

56.   (Previously Presented) The storage medium of claim 53, wherein said program instructions, upon execution, cause said processor to cost each said driver identified by said user.

57.   (Currently Amended) A computer-implemented method of managing a process, said computer implemented method comprising:

identifying activities that comprise the process, wherein the process is a series of activities, wherein an input of at least one subsequent activity is dependent on an output of at least one previous activity;

identifying drivers associated with at least one metric, reflecting an efficiency of said process, for each of the activities, wherein identifying said drivers includes identifying at least one of fixed components and variable components of each said driver;

determining, in a computer, a metric for each said driver based on said at least one of fixed components and variable components thereof;

generating, in a computer, a model of said process at least as a

function of said drivers by combining representations of activities comprising said

process; and

outputting, from said model in a computer, a predictive metric reflecting

an efficiency of the total process.

58.   (Previously Presented) A method of claim 57, wherein the fixed and

variable components of each said driver are the fixed and variable costs of each said

driver.